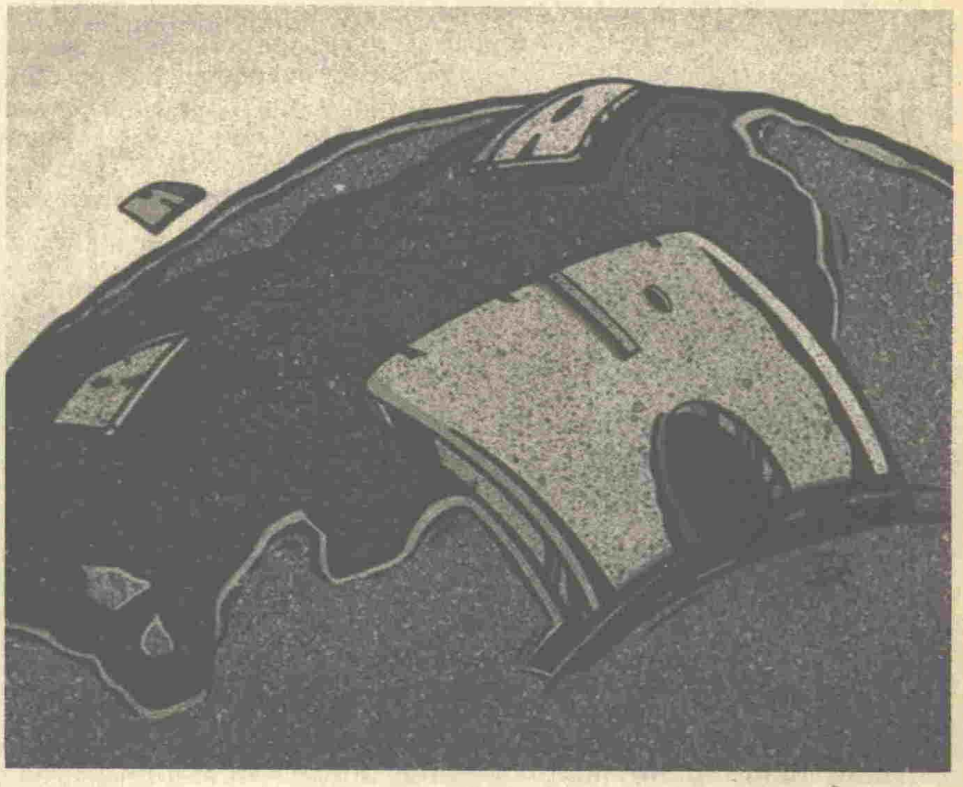
SOFT MAIL

DIET
บีบแล้วใช้ได้เลย

ลดความอ้วนให้โปรแกรม
เพื่อเพิ่มเนื้อที่บนดิสค์
ด้วยโปรแกรมยูทิลิตี้



หลาย ๆ ท่านคงจะเคยได้ลองใช้โปรแกรมลดขนาดข้อมูลอย่างเช่น PkARC, Lharc หรือ PkZIP ที่โด่งดังมากในบ้านเรา การใช้โปรแกรมเหล่านี้จะช่วยให้เราสามารถประหยัดเนื้อที่ในการเก็บข้อมูลและโปรแกรมไปได้มากทีเดียว เพียงแต่เราต้องมาขยายข้อมูลให้กลับเป็นปกติก่อนนำข้อมูลเหล่านั้นมาใช้งานเท่านั้น เมื่อเราได้ลองใช้โปรแกรมลดขนาดข้อมูลเหล่านี้ดูแล้ว ทุกคนมักตอบเสียงเดียวกันว่ายอดเยี่ยม เพราะเราสามารถที่จะเก็บข้อมูลและโปรแกรมหลาย ๆ ตัวไว้ในฮาร์ดดิสค์โดยเสียเนื้อที่น้อยลง เช่น จากของเดิมต้องเสียเนื้อที่ 1.5 เมกะไบต์ เราก็จะบีบขนาดให้เล็กลง ซึ่งอาจจะเหลือเพียง 800 กิโลไบต์โดยเฉลี่ย (อาจจะมากกว่าหรือน้อยกว่านี้บ้าง ตามแต่ประเภทของไฟล์) ซึ่งหากเราเก็บโปรแกรมหลาย ๆ ตัวแล้ว มีพื้นที่ส่วนกลางเอาไว้ขยายข้อมูลเมื่อจำเป็นต้องใช้และบีบข้อมูลเก็บไว้แบบเดิมหลังจากใช้งานเสร็จแล้ว เราก็สามารถจะเก็บโปรแกรมได้เท่าเดิมโดยใช้เนื้อที่ในการเก็บข้อมูลน้อยลงหรือสามารถเก็บอะไรได้มากขึ้นในเนื้อที่เท่าเดิม

มาถึงตรงนี้ทุกคนคงจะเป็นประโยชน์ของโปรแกรมเหล่านี้แล้ว อย่างไรก็ตามมนุษย์เองก็ยังคงหาสิ่งที่ดีกว่าอยู่ร่ำไป จึงทำให้มีโปรแกรมบีบขนาดข้อมูลที่มีความสามารถมากเกิดขึ้น และหนึ่งในนั้นก็คือ 'DIET'

DIET เกิดขึ้นมาเพื่อแก้ไขความไม่สะดวกสบายในการใช้งาน PkZIP ให้หมดไป คือ โดยปกติเรามีความจำเป็นต้องขยายข้อมูลที่บีบเอาไว้ก่อนนำมาใช้งานด้วยโปรแกรม PkunZIP แต่สำหรับโปรแกรมที่เราทำการบีบให้เล็กลงด้วย DIET นั้นจะยังคงความสามารถเรียกใช้ได้โดยตรงจาก DOS Prompt ได้ทันทีเหมือนเดิมทุกประการและต่อจากนี้ไปเราจะกล่าวถึงรายละเอียดของโปรแกรมประเภทนี้

ประวัติ

ก่อนที่ DIET จะเกิดขึ้นมาบนโลกนี้ ได้มีโปรแกรมประเภทเดียวกันนี้เกิดขึ้นมาก่อนแล้ว และถ้าจะนับอันดับ DIET จะเป็นน้องใหม่ล่าสุดที่มีความสามารถมากเลยทีเดียว DIET มีพี่อยู่สองตัวคือ LZEXE และ PkLite พี่ใหญ่คือ LZEXE ซึ่งเป็นแบบฉบับของโปรแกรมเหล่านี้เลยทีเดียว น่าเสียดาย

ว่าคนเขียนโปรแกรมนี้เป็นชาวฝรั่งเศส และทำเป็นเวอร์ชันที่ใช้ภาษาฝรั่งเศสเท่านั้น เรื่องรายละเอียดไม่ต้องพูดถึงเพราะอ่านไม่ออกเลยครับ คู่มือเป็นภาษาฝรั่งเศสหมด ขนาดโปรแกรมจะถามว่า Y/N ยังใช้เป็น O/N เลยครับ ปัญหาอีกอย่างของ LZEXE คือไม่สามารถที่จะคืนสภาพไฟล์ที่บีบข้อมูลไปแล้วได้ แม้ว่าตอนหลังจะมีผู้เขียนโปรแกรมเพื่อคืนสภาพข้อมูลที่ถูกบีบขนาดข้อมูลด้วย LZEXE แล้วก็ไม่ทำให้ LZEXE ได้รับความนิยมมากขึ้น เพราะข้อมูลที่ถูกคืนสภาพไม่เหมือนเดิม 100% อีกทั้งเป็นภาษาฝรั่งเศส

หลังจากที่ปล่อยให้ LZEXE เป็นพระเอกอยู่เป็นปี ทาง Pkware ซึ่งเป็นยักษ์ใหญ่ในวงการโปรแกรมที่ใช้ในการบีบขนาดข้อมูลที่มีผลงานเป็นที่ยอมรับกันทั่วโลก ไม่ว่าจะเป็นมาตรฐาน ARC หรือ ZIP ซึ่งกลายเป็นมาตรฐานที่ทุกคนยอมรับ ในปัจจุบันจึงได้ออกโปรแกรม PkLite รุ่นแรกสู่สายตาชาวคอมพิวเตอร์ ซึ่งเป็นเพียงเบต้าเวอร์ชันเท่านั้น มีความสามารถเหมือน LZEXE แต่เร็วกว่า สามารถที่จะคืนไฟล์ได้ในสภาพเดิมสมบูรณ์กว่า LZEXE มากและที่สำคัญคือทุกอย่างเป็นภาษาอังกฤษ ทำให้ PkLite เป็นที่นิยมอย่างรวดเร็ว และไม่นาน Pkware ก็ออกเวอร์ชันใหม่ตามออกมาหลังจากที่ประกาศตัวเบต้าเวอร์ชันไม่นานนัก ในเวอร์ชันใหม่นี้มีความสามารถมากขึ้นและมีความสามารถบางอย่างที่เป็นประโยชน์ต่อบริษัทเขียนซอฟต์แวร์คือการสั่งให้ไฟล์ที่ถูกบีบข้อมูลไม่สามารถขยายออกมาในสภาพเดิมได้ ไม่ว่าจะกรณีใดก็ตาม ซึ่งจะทำให้คนที่ชอบแกะโปรแกรมไม่สามารถที่จะทำการแก้ไขข้อมูลที่ถูกบีบขนาดเอาไว้ได้ ทำให้ซอฟต์แวร์เฮ้าส์หลายแห่ง (ในต่างประเทศ) ได้นำ PkLite มาใช้กับซอฟต์แวร์ที่ตัวเองผลิตขึ้น ทั้งนี้รวมถึงเวิร์ดจุพารุ่นหลัง ๆ ที่มีขนาดเล็กลงก็มาจากการนำ PkLite มาใช้เหมือนกัน ซึ่งก็เป็นช่วงเดียวกับที่มีการเผยแพร่ DIET รุ่นแรกออกมา DIET เขียนขึ้นโดยชาวญี่ปุ่น แต่ใช้ภาษาอังกฤษในการสื่อสารกับผู้ใช้ ในเวอร์ชันแรก DIET มีความสามารถเหมือนกับ PkLite เกือบทุกอย่าง ทำให้ DIET ไม่สามารถเข้าสู่ความนิยมของผู้ใช้ได้มากเท่ากับ PkLite และแล้ว PkLite ก็ได้ออกเวอร์ชัน 1.12 ออกมา มีความสามารถในการบีบข้อมูลได้เร็วกว่าเวอร์ชันเก่า ขนาดเล็กกว่าเล็กน้อย แต่ข้อผิดพลาดที่ใหญ่หลวงของเวอร์ชันนี้ก็ปรากฏออกมาด้วยคือ PkLite รุ่นนี้ไม่มีความสามารถที่จะคืนสภาพโปรแกรมที่ถูกบีบด้วย PkLite เวอร์ชันเก่า ๆ ได้ เหตุการณ์ที่เกิดขึ้นคือมันจะคืนสภาพโปรแกรมที่ถูกบีบขนาดเอาไว้โดย PkLite เวอร์ชันเก่าออกมาเป็นไฟล์อะไรก็ไม่ทราบ ตัวโปรแกรมที่ถูกคืนสภาพจะเสียหายทันที นั่นหมายความว่าผู้ที่กำลังใช้ PkLite เวอร์ชันเก่าอยู่จะต้องระวังอย่างมาก หากต้องการคืนสภาพโปรแกรมที่ถูกบีบขนาดเอาไว้ไม่เช่นนั้นอาจจะต้องเสียโปรแกรมนั้น ๆ ไป จากจุดนี้เองทำให้ DIET เวอร์ชัน 1.05 ที่เปิดตัวในเวลาไล่เลี่ยกันเริ่มเข้ามาแทนที่พร้อมด้วยความสามารถใหม่ ๆ ที่ PkLite ไม่มี เช่นทำงานในแบบภาษาญี่ปุ่น, มีการใช้อัลกอริทึมที่เน้นในเรื่องความเร็วในการทำงานและอื่น ๆ อีกมากพอสมควร

และที่กล่าวมาก็คือ ประวัติเกือบทั้งหมดที่ทราบ มาถึงจุดนี้ผมคิดว่าคงจะมีข้อมูลให้ท่านผู้อ่านตัดสินใจในการใช้โปรแกรมตัวใดตัวหนึ่งมากพอสมควร เพราะ PkLite เองก็มาจากยักษ์ใหญ่ Pkware พร้อมทั้งมีความสามารถที่ไม่อนุญาตให้คืนขนาดโปรแกรมที่ถูกบีบข้อมูล ส่วน DIET เองก็มาจากนักเขียนโปรแกรมชาวอาทิตย์อุทัย ซึ่งมีออปชันแปลก ๆ มากกว่า PkLite พอสมควร

จุดเด่นของ DIET

ในช่วงนี้เราจะพูดถึงความสามารถของ DIET กัน

- DIET ถูกสร้างขึ้นเพื่อลดขนาดไฟล์โปรแกรมประเภท .COM และ .EXE ตามที่ PkLite ทำได้ แต่สำหรับเวอร์ชันนี้ DIET ได้เพิ่มความสามารถในการลดขนาดของไฟล์ทุกชนิด รวมถึงไฟล์ข้อมูล เช่น เท็กซ์ไฟล์ ด้วย
- สำหรับท่านที่ใช้ JDOS ของ IBM Japan อยู่ สามารถที่จะสั่งให้ DIET โต้ตอบออกมาเป็นภาษาญี่ปุ่นได้
- DIET รุ่นนี้ใช้อัลกอริทึมแบบใหม่ซึ่งเน้นทางด้านความเร็ว ดังนั้น DIET จึงเป็นโปรแกรมที่ทำงานเร็วที่สุดในบรรดาโปรแกรมประเภทเดียวกันนี้
- DIET สามารถที่จะคืนสภาพข้อมูลที่ถูกบีบเอาไว้ได้สมบูรณ์ 100% รวมถึงสามารถคืนสภาพข้อมูลที่ถูกบีบขนาดไว้ด้วย DIET รุ่นเก่าได้เช่นกัน ดังนั้นจึงแน่ใจในความปลอดภัยที่จะเปลี่ยนเวอร์ชันใหม่ได้เลย
- หาก DIET พบว่าไฟล์ที่ถูกบีบขนาดแล้ว ต้องใช้จำนวนคลัสเตอร์ของดิสก์เท่ากันกับไฟล์ต้นฉบับ DIET จะคงสภาพไฟล์ต้นฉบับไว้อย่างเดิม เช่น หากคลัสเตอร์ของเรามีขนาด 8 กิโลไบต์ แล้วโปรแกรมเดิมมีขนาด 16 กิโลไบต์ หรือ 2 คลัสเตอร์ ให้ DIET จัดการลดขนาดข้อมูลแล้วข้อมูลที่ได้อาจจะมีขนาด 11 กิโลไบต์ ซึ่งในกรณีนี้ดอสจะใช้คลัสเตอร์ในการเก็บเท่ากัน DIET ก็จะคงสภาพโปรแกรมต้นฉบับเอาไว้ เพราะสามารถเรียกขึ้นมาทำงานได้เร็วกว่า

การใช้งาน

DIET มีการใช้งานอยู่ 2 แบบด้วยกันคือ แบบเรสซิเดนต์และแบบนอนเรสซิเดนต์จะขอพูดถึงการทำงานในแบบนอนเรสซิเดนต์ก่อน การทำงานในแบบนอนเรสซิเดนต์นั้นจะเหมือนกับโปรแกรมอื่น ๆ คือเรียกโปรแกรม DIET ตามชื่อไฟล์ที่ต้องการบีบขนาด มีรูปแบบการเรียกใช้ดังนี้

```
C:>DIET <filename.exe>
```

สำหรับชื่อไฟล์ที่จะใส่แทน <filename.exe> สามารถที่จะใช้เป็น Wild Card Character เช่น * หรือ ? ได้ เช่น *.* ก็จะหมายความถึงทุก ๆ ไฟล์ในไดเรคทอรีปัจจุบัน หรือไม่ก็อาจจะใส่พร้อมกันทีเดียวสองชื่อเลยก็ได้เช่น

```
C:>DIET file1.com file2.txt
```

ผลที่ได้จากคำสั่งนี้คือ DIET จะทำการย่อขนาดของโปรแกรม file1.com และไฟล์ข้อมูล file2.txt พร้อมกัน หรืออาจจะใช้เป็น

```
C:>DIET *.COM *.EXE *.DAT
```

ซึ่ง DIET ก็จะทำการย่อขนาดข้อมูลทุกไฟล์ที่มีนามสกุล เป็น .COM .EXE และ .DAT เช่นกัน

การทำงานในแบบเรสซิเดนต์นั้นเป็นจุดเด่นจุดหนึ่งที่มีเฉพาะ DIET เท่านั้น โดยปกติไฟล์โปรแกรมที่บีบขนาดโดย DIET หรือ PkLite ก็ตาม จะมีการเพิ่มส่วนหัวเข้าไปในไฟล์ที่ถูกบีบขนาดเอาไว้ เพื่อที่จะให้โปรแกรมใส่ไว้ในส่วนหัวขยายข้อมูลให้กลับสู่สภาพปกติในหน่วยความจำ แล้วจึงกระโดดไปที่จุดเริ่มต้นของโปรแกรมนั้น ๆ เพื่อให้โปรแกรมเหล่านั้นทำงานตามปกติ คราวนี้ DIET สามารถที่จะลดขนาดไฟล์ที่เป็นข้อมูล ซึ่งไฟล์เหล่านี้ไม่สามารถเรียกขึ้นมาทำงานได้ จึงไม่สามารถที่จะใส่ส่วนขยายข้อมูลไว้ที่ตอนต้นของไฟล์ที่ถูกบีบขนาดเอาไว้ได้ DIET จึงได้แก้ปัญหาในจุดนี้คือ สร้างตัวเองให้ทำงานในแบบเรสซิเดนต์ได้ เมื่อเราเรียกให้ DIET เข้าไปฝังตัวอยู่ในหน่วยความจำแล้ว DIET จะคอยตรวจสอบว่าไฟล์ข้อมูลที่ถูกเปิดขึ้นนั้น ไฟล์ใดที่ถูกบีบขนาดเอาไว้ด้วย DIET บ้าง หากพบจะทำหน้าที่ขยายขนาดข้อมูลคืนตามปกติแล้วจึงค่อยส่งข้อมูลที่ถูกขยายแล้วไปให้โปรแกรมที่เรียกใช้ข้อมูลนั้น ๆ อีกที ข้อจำกัดของการทำงานกับข้อมูลที่ถูกบีบขนาดด้วย DIET คือ เราไม่สามารถแก้ไขข้อมูลได้ หรือหากว่าข้อมูลมีการแก้ไข DIET ก็จะสร้างไฟล์ใหม่ขึ้นมาเพื่อเก็บข้อมูลที่ถูกแก้ไขไว้ในรูปแบบของไฟล์ตามปกติที่ไม่มีการบีบข้อมูลแต่อย่างใด ถ้าผู้อ่านยังไม่เข้าใจการทำงานแบบเรสซิเดนต์ของ DIET ก็สามารถดูรูปประกอบเปรียบเทียบการใช้โปรแกรมดีบักของดอสเรียก TXT ไฟล์ที่ถูกบีบเอาไว้ด้วย DIET ออกมาดูในขณะที่มีและไม่มีการเรียก DIET ให้ฝังตัวอยู่ในหน่วยความจำของคอมพิวเตอร์ จึงแสดงได้ดังรูปที่ 1-2 หากเครื่องของเราติดตั้งหน่วยความจำระบบ EMS แล้วโปรแกรม DIET จะย้ายตัวเองไปอยู่ใน EMS ทำให้ส่วนที่จำเป็นต่อการเรียกใช้

```
C:\TEMP>debug ic.doc
-d
20E3:0100  9D 89 64 6C 7A 00 49 FC-D3 8B 0C DE 0D C3 75 0D   ..dlz.I.......u.
20E3:0110  0A FE FC F8 D8 81 F0 3B-38 E6 09 20 20 5C 87 FE   .......;8..  \..
20E3:0120  FC F8 1D D0 F4 49 FF FF-6E 74 65 6C 69 43 61 74   .....I..nteliCat
20E3:0130  28 74 6D 29 20 44 69 73-FD EF 6B 20 F3 61 6C 6F   (tm) Dis..k .alo
20E3:0140  67 65 72 C7 E0 FF 06 22-53 6F 66 74 77 61 72 65   ger....."Software
20E3:0150  F8 1F 20 F7 6C 75 74 69-6F 6E 73 20 74 68 CB EF   .. .lutions th..
20E3:0160  63 20 6D 61 6B EC 65 F1-65 22 D7 C1 B2 86 09 7F   c mak.e.e".......
20E3:0170  FC 43 6F 70 79 72 69 67-68 CD 28 63 29 31 39 FF   .Copyrigh.(c)19.
-d
20E3:0180  18 39 30 2D 39 31 2C 20-42 79 E7 75 72 B0 F8 1D   .90-91, By.ur...
20E3:0190  B2 4C 69 74 74 6C 65 48-74 7E 0C FA 41 6C 6C 20   .LittleHt~..All
20E3:01A0  52 17 E3 B8 CD 52 65 8A-72 76 15 3C 65 64 E5 2F   R....Re.rv.<ed./
20E3:01B0  46 6F 72 FF 63 EC 42 4D-20 50 43 73 6E 58 54 DF   For.c.BM PCsnXT.
20E3:01C0  7D FB 41 FB 50 53 2F 32-FC EF 9B 61 6E 64 20 31   }.A.PS/2...and 1
20E3:01D0  30 30 25 20 33 3D C7 88-6D 70 0D 69 62 59 73 BB   00% 3=..mp.ibYs.
20E3:01E0  0A 3A 0C 8B 1E 28 B8 CF-0E 54 61 A9 20 9B 7B 0D   .:...(...Ta. .{.
20E3:01F0  9B 62 FD 07 1D 9E 03 D4-E6 AB D5 63 8F C3 75 20   .b.........c..u
-q
```

รูปที่ 1 แสดงตัวโปรแกรมในขณะที่ยังไม่ได้บีบข้อมูล

```
C:\TEMP>diet -z
<<<< exciting disk space saver DIET version 1.02b >>>>
    ... copyright(c) 1991   by Teddy Matsumoto ...

Stayed on memory!

C:\TEMP>debug ic.doc
-d
257F:0100  0D 0A 0D 0A 0D 0A 0D 0A-0D 0A 0D 0A 0D 0A 0D 0A  ................
257F:0110  0D 0A 0D 0A 0D 0A 0D 0A-0D 0A 0D 0A 0D 0A 0D 0A  ................
257F:0120  0D 0A 0D 0A 0D 0A 0D 0A-0D 0A 0D 0A 0D 0A 0D 0A  ................
257F:0130  0D 0A 0D 0A 0D 0A 0D 0A-0D 0A 20 20 20 20 20 20  ..........
257F:0140  20 20 20 20 20 20 20 20-20 20 20 20 20 20 20 20
257F:0150  20 20 20 20 20 49 6E 74-65 6C 69 43 61 74 28 74      InteliCat(t
257F:0160  6D 29 20 44 69 73 6B 20-43 61 74 61 6C 6F 67 65  m) Disk Cataloge
257F:0170  72 0D 0A 20 20 20 20 20-20 20 20 20 20 20 20 20  r..
-d
257F:0180  20 20 20 20 20 20 20 20-22 53 6F 66 74 77 61 72          "Softwar
257F:0190  65 20 53 6F 6C 75 74 69-6F 6E 73 20 74 68 61 74  e Solutions that
257F:01A0  20 6D 61 6B 65 20 53 65-6E 73 65 22 28 74 6D 29   make Sense"(tm)
257F:01B0  0D 0A 0D 0A 20 20 20 20-20 20 20 20 20 20 20 20  ....
257F:01C0  20 20 20 20 20 20 20 20-20 43 6F 70 79 72 69          Copyri
257F:01D0  67 68 74 20 28 63 29 31-39 39 30 2D 39 31 2C 20  ght (c)1990-91,
257F:01E0  42 79 20 43 75 72 74 69-73 20 4C 69 74 74 6C 65  By Curtis Little
257F:01F0  0D 0A 20 20 20 20 20 20-20 20 20 20 20 20 20 20  ..
-q

C:\TEMP>
```

รูปที่ 2 แสดงโปรแกรมหลังจากการทำการบีบข้อมูลไว้แล้ว

จริง ๆ เท่านั้นที่อยู่ในส่วนของหน่วยความจำหลัก ทำให้มีหน่วยความจำเหลือใช้มากขึ้น

คำสั่งของ DIET

จริง ๆ แล้ว DIET ไม่มีความจำเป็นที่จะต้องเรียนรู้คำสั่งมากมาย เพราะโดยพื้นฐานเมื่อเรียกใช้งาน DIET พร้อมชื่อไฟล์ DIET ก็จะรู้หน้าที่ว่าจะต้องทำการบีบขนาดข้อมูลของไฟล์ที่ระบุเอาไว้อยู่แล้ว ฉะนั้นเนื้อเรื่องในส่วนนี้จะเป็นการเรียนรู้คำสั่งต่าง ๆ ที่มีอยู่ใน DIET เพื่อการใช้งานที่เต็มประสิทธิภาพที่สุด ในการใช้คำสั่งพารามิเตอร์ (Parameter) ที่จะกล่าวถึงต่อไปนี้นั้น โดยปกติจะใช้เครื่องหมาย "-" นำหน้าเพื่อบอก DIET ว่าเป็นการใช้พารามิเตอร์ แต่หากใครรู้สึกเคยชินกับเครื่องหมาย "/" ก็สามารถจะใช้เครื่องหมายนี้ได้เหมือนกัน

รูปแบบการใช้พารามิเตอร์โดยทั่วไปคือ

```
DIET [Parameter]
[filename. exe]
```

คำสั่งทั้งหมดสามารถแบ่งเป็นประเภทได้ดังนี้

คำสั่งเกี่ยวกับการบีบข้อมูล : หาก DIET พบว่าเราไม่มีคำสั่ง -l, -r, -ra, -z, -za, -zd และ -c แล้ว DIET จะเริ่มทำการบีบขนาดข้อมูลของไฟล์ที่ระบุทันที

คำสั่งในการขยายข้อมูลกลับเหมือนเดิม : -r, -ra

คำสั่งในการฝังตัวในหน่วยความจำ : -z, -za, -zd

คำสั่งในการดูรายชื่อเฉพาะไฟล์ที่ถูกบีบขนาดข้อมูลไว้ และคำสั่งในการตรวจสอบไฟล์ที่ถูกบีบข้อมูลไว้ : -l, -c

และมีคำสั่งจิปาถะที่เหลือคือ : -o, -t, -i, -m และ -a

คำสั่งในการบีบข้อมูล

DIET จะทำการบีบข้อมูลทันทีที่ไม่พบพารามิเตอร์ -l, -r, -ra, -z, -za, -zd และ -c มีจุดเพิ่มเติมคือ โดยปกติ DIET จะสร้างไฟล์ที่ถูกบีบข้อมูลขึ้นมาทับไฟล์เดิม ซึ่งเราสามารถจะระบุให้ DIET สร้างแบคอัพไฟล์ให้เราได้โดยการใช้พารามิเตอร์ -o และในการบีบข้อมูลบางครั้งอาจจะไม่ประสบผลสำเร็จ ไม่ต้องตกใจนะครับ เพราะ

เหตุการนี้จะเกิดขึ้นได้ต่อเมื่อ

1. ไฟล์ที่ถูกบีบขนาด + ส่วนหัวที่จะเพิ่มเข้าไปนั้น มีขนาดใหญ่กว่าไฟล์เดิม

2. ถึงแม้ว่าไฟล์ใหม่จะมีขนาดเล็กลงแต่ยังคงคิดจำนวนคลัสเตอร์เท่าเดิม

3. ไฟล์นั้นเป็น .EXE ที่มี Overlay รวมอยู่ในไฟล์เดียวกัน (สำหรับข้อ 3 นี้มีพารามิเตอร์ -i เพื่อบังคับให้ DIET ยังคงบีบขนาดไฟล์ประเภทนี้ สามารถดูรายละเอียดได้ที่คำอธิบายเกี่ยวกับพารามิเตอร์ i)

การคืนสภาพให้เป็นปกติ

พารามิเตอร์ที่จะใช้มีอยู่สองตัว คือ −r และ −ra มีหน้าที่ในการคืนสภาพข้อมูลเหมือนกันทุกประการ มีข้อแตกต่างคือ พารามิเตอร์ −ra จะขยายไฟล์ทุกไฟล์ที่ระบุ แต่พารามิเตอร์ −r จะทำการขยายข้อมูลเฉพาะไฟล์ที่มีนามสกุล .COM และ .EXE หากเรามีไฟล์ TEXT.TXT, TEST.COM, TEST.EXE อยู่ใน Current Directory แล้วใช้คำสั่ง

```
C:>DIET −r *.*
```

ไฟล์ที่ DIET จะคืนสภาพให้จะเป็นไฟล์ TEST.COM และ TEST.EXE เท่านั้น แต่ถ้าหากเปลี่ยนพารามิเตอร์จาก −r เป็น −ra ไฟล์ทุกไฟล์รวมทั้ง TEST.TXT จะถูกคืนสภาพให้เป็นแบบเดิมทั้งหมด

การทำงานในแบบฝังตัว

การทำงานในแบบนี้จะใช้พารามิเตอร์ −z ซึ่งจะทำงานสองหน้าที่คือ บอกให้ DIET ฝังตัวเองในหน่วยความจำ และยังทำหน้าที่สั่งให้ DIET เอาตัวเองออกจากหน่วยความจำด้วยในกรณีที่มีการตรวจพบว่า ได้มี DIET ฝังตัวอยู่ในหน่วยความจำอยู่ก่อนแล้ว หากมีการใช้โปรแกรม DIET ในการลดขนาดข้อมูลของไฟล์ที่เป็นข้อมูลหรือไฟล์ที่มีนามสกุลไม่ใช่ .COM และ .EXE ขอแนะนำว่าให้ใส่คำสั่งนี้ใน AUTOEXEC.BAT เลยดีกว่า เราสามารถใส่ตัวเลขหลังพารามิเตอร์ −z ได้ เพื่อใช้ในการกำหนดหน่วยความจำที่จะใช้ขยายข้อมูลให้กลับเป็นปกติก่อน แล้วจึงส่งข้อมูลที่คืนสภาพเสร็จเรียบร้อยแล้วไปให้กับโปรแกรมที่เรียกใช้ข้อมูล โดยตัวเลขสามารถมีค่าได้ตั้งแต่ 1 ถึง 9 หากไม่มีการระบุจะมีค่าเป็น 2 มีรูปแบบการใช้ดังนี้

```
C:>DIET -z8
```

การใส่ตัวเลขยิ่งมากจะทำให้ DIET สามารถทำงานได้เร็วขึ้น แต่ก็ต้องใช้หน่วยความจำมากขึ้นเช่นกัน

ในบางครั้งต้องการที่จะยกเลิกการใช้งาน DIET ชั่วคราว เราสามารถที่จะใช้พารามิเตอร์ −zd เพื่อ Disable หรือระงับการทำงานของ DIET ชั่วคราว โดยไม่ต้องนำ DIET ออกจากหน่วยความจำ รูปแบบคำสั่งคือ

```
C:>DIET −zd
```

หากต้องการให้ DIET กลับมาทำงานตามปกติ สามารถทำได้โดยใช้พารามิเตอร์ −za ซึ่งจะมีรูปแบบในการเรียกใช้เหมือนกับพารามิเตอร์ −zd ทุกอย่าง แต่การนำ DIET ออกจากหน่วยความจำไปเลยก็ใช้คำสั่ง −z เหมือนกับตอนที่สั่งให้ DIET ฝังตัวในหน่วยความจำ ข้อควรระวังในการใช้พารามิเตอร์ −z คือ การใช้พารามิเตอร์นี้เพื่อนำ DIET ออกจากหน่วยความจำหลังจากมีการเรียกเอาโปรแกรมเรสซิเดนต์ตัวอื่น ๆ เข้าไปไว้ในหน่วยความจำต่อจากตัวโปรแกรม DIET อาจจะเกิดปัญหาขึ้นได้ หากพบว่าโปรแกรมนั้นมีการใช้หรือไปเปลี่ยนแปลงอินเตอร์รัพต์ตัวเดียวกับที่ DIET นำมาใช้ ถ้ามีความจำเป็นที่จะต้องเอา DIET ออกจากหน่วยความจำจริง ๆ แล้วขอแนะนำให้ใช้โปรแกรม Mark และ Release จะดีกว่า ซึ่งมีให้เลือกใช้มากมาย แต่ที่นิยมใช้เห็นจะเป็นของ Turbo Power (บริษัทเดียวกับที่ขาย Btree Filer 5.0)

การเรียกชื่อไฟล์ที่ถูกบีบ

คำสั่งที่ใช้คือ

```
C:>DIET −l <filename.ext>
```

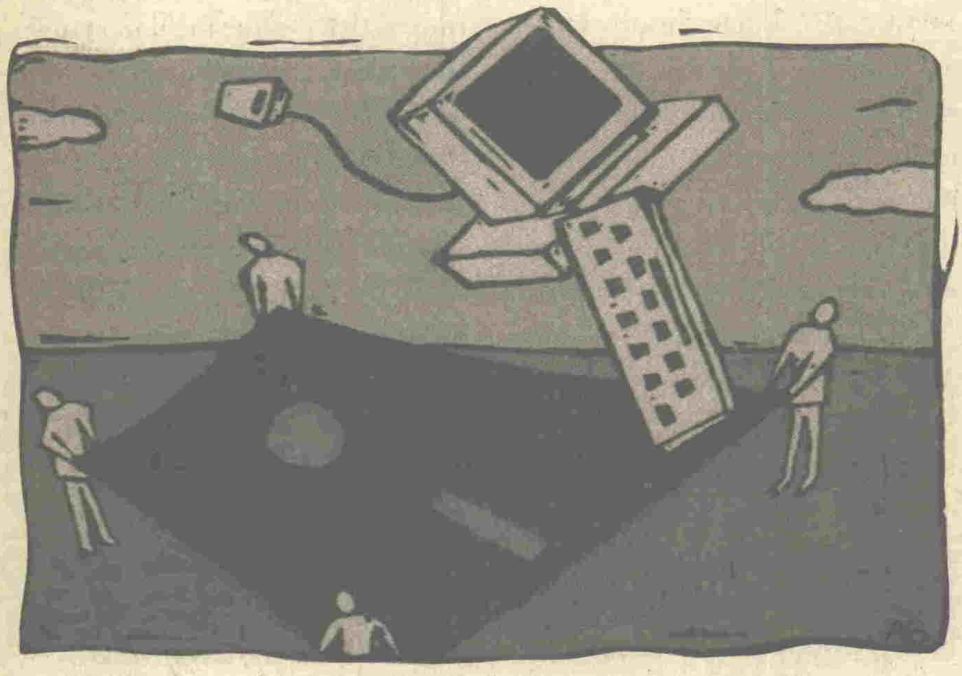
คำสั่งนี้จะแสดงรายชื่อไฟล์ที่ถูกบีบข้อมูลด้วย DIET ในไดเรคทอรีนั้น ๆ ออกมาดู หากต้องการดูรายชื่อไฟล์ทุก ๆ ไฟล์ที่ถูกบีบขนาดข้อมูลด้วย DIET สามารถใช้เพียงพารามิเตอร์ −l โดยไม่ต้องใส่ชื่อไฟล์ก็ได้

ตรวจสอบความเรียบร้อยของไฟล์

คำสั่งที่ใช้คือ

```
C:>DIET −c <filename.ext>
```

การใช้งานจะเหมือนกับพารามิเตอร์ −l ทุกอย่าง แต่ไม่สามารถละชื่อไฟล์ที่จะต้องใส่ตามพารามิเตอร์ −c เท่านั้น คำสั่ง DIET จะทำการตรวจสอบไฟล์ที่ถูกบีบขนาดข้อมูลด้วย DIET ว่ายังมีความถูกต้องครบถ้วนหรือเปล่า เนื่องจากไฟล์ที่ถูกบีบข้อมูลเอาไว้นั้นแม้จะมี



การเปลี่ยนแปลงไปเพียง 1 บิต ก็อาจจะทำให้เกิดความเสียหายและไม่สามารถเรียกออกมาใช้งานได้เลยก็ได้ แต่เหตุการณ์แบบนี้โดยปกติจะไม่เกิดขึ้นเลย หากไม่เข้าไปแก้ไขไฟล์เหล่านี้โดยตรงด้วยโปรแกรมประเภทนอร์ตันยูทิลิตี้ หรือพีซีทูลส์โดยไม่มีการเรียก DIET ให้ฝังตัวอยู่ในหน่วยความจำก่อน

การกำหนด Output

หากจำเป็นต้องการเก็บต้นฉบับของข้อมูลหรือโปรแกรมที่จะถูกบีบขนาด หรือต้องการสร้างไฟล์ที่ถูกบีบเอาไว้ในไดเรคทอรีอื่น ที่กำหนดขึ้นเป็นพิเศษ เราสามารถสั่งงานได้โดยใช้พารามิเตอร์ –o ดังตัวอย่างต่อไปนี้

<ตัวอย่างที่ 1.> เมื่อเราอยู่ที่ไดรฟ์อื่น ๆ หรือไดเรคทอรีในฮาร์ดดิสก์ซึ่งไม่ใช่ไดรฟ์ A: แต่ต้องการที่จะบีบขนาดข้อมูลซึ่งต้นฉบับบันทึกอยู่ในไดรฟ์ A: สมมติให้ชื่อว่า TEST.DAT โดยสร้างแฟ้มข้อมูลที่ถูกบีบขนาดเรียบร้อยแล้วที่ไดรฟ์หรือไดเรคทอรีปัจจุบันที่เรากำลังอยู่ สามารถทำได้โดยใช้คำสั่ง

```
C:>DIET –o A:TEST.DAT
```

<ตัวอย่างที่ 2.> เป็นกรณีที่กลับกันกับตัวอย่างที่ 1 นั้นคือเราต้องการบีบขนาดของข้อมูลที่อยู่ในไดรฟ์หรือไดเรคทอรีปัจจุบันไปยังไดรฟ์อื่นหรือไดเรคทอรีอื่น โดยสมมติให้ไฟล์ที่นำมาทดสอบชื่อ TEST.DAT สามารถทำได้โดย

```
C:>DIET –oA: TEST.DAT หรือ
C:>DIET –oC:\TEMP\TEST.DAT
```

คำสั่งในบรรทัดแรกจะได้ไฟล์ TEST.DAT ที่บีบขนาดข้อมูลแล้วเก็บไว้ที่ไดรฟ์ A: หากเป็นคำสั่งบรรทัดที่สอง เราจะได้ไฟล์ TEST.DAT ที่ถูกบีบขนาดข้อมูลเรียบร้อยแล้วเช่นเดียวกันอยู่ในไดเรคทอรี C:\TEMP. จากตัวอย่างในบรรทัดที่ 2 จะเห็นว่าถ้าต้องการใช้ DIET เก็บข้อมูลที่ถูกบีบขนาดเอาไว้ที่ไดเรคทอรีใดนั้นระบุ "\" ปิดท้ายไดเรคทอรีทุกครั้ง

กำหนดไดเรคทอรีหรือไดรฟ์

ในการทำงานแบบเรสซิเดนต์ DIET จะสร้างไฟล์ชั่วคราวขึ้นมา ซึ่งโดยปกติจะสร้างไว้ในไดเรคทอรีปัจจุบัน เราสามารถที่จะกำหนดไดรฟ์หรือไดเรคทอรีที่แน่นอนเพื่อให้ DIET ไปสร้างไฟล์ชั่วคราวไว้ที่นั่นได้ด้วยการใช้พารามิเตอร์ –t เช่น

```
C:>DIET –z –td:\
```

โดยคำสั่งนี้จะสั่งให้ DIET ฝังตัวในหน่วยความจำและสร้างไฟล์ชั่วคราวขึ้นที่รูทไดเรคทอรีของไดรฟ์ D: ซึ่งไดรฟ์ D: อาจจะเป็นแรมไดรฟ์ที่เราสร้างขึ้นก็ได้ (หากต้องการระบุชื่อไดเรคทอรีโปรดอย่าลืมว่าต้องใส่เครื่องหมาย "\" ปิดท้ายชื่อไดเรคทอรีด้วยทุกครั้ง) หากเครื่องที่ใช้งานอยู่นั้นมีหน่วยความจำประเภท EMS ติดตั้งอยู่มากเพียงพอแล้ว DIET ก็จะไปสร้างไฟล์ชั่วคราวในหน่วยความจำแทน และวิธีนี้จะทำให้ DIET ทำงานได้เร็วและดูราบรื่นเหมือนการทำงานในแบบนอนเรสซิเดนต์มากกว่า

ยกเลิกคำเตือนของ DIET

หากมีการสั่งให้ DIET ลดขนาดไฟล์ .EXE ที่มีโอเวอร์เลย์ในตัวอาจจะมีปัญหาได้ ไฟล์พวกนี้โดยปกติโปรแกรมจะสร้างไฟล์ .EXE และไฟล์ .OVL แยกกัน แต่บางโปรแกรมจะรวมทั้งสองไฟล์อยู่ในไฟล์ .EXE เพียงไฟล์เดียวเท่านั้น สามารถสังเกตโปรแกรมเหล่านี้ได้ง่ายมาก คือ ไฟล์ .EXE จะมีขนาด

ใหญ่ แต่เวลาเรียกขึ้นมาทำงานค่อนข้างจะเร็วกว่าปกติมาก โปรแกรมพวกนี้บางตัวเมื่อนำมาบีบขนาดด้วย DIET แล้ว จะมีปัญหาการเรียกใช้งาน ดังนั้นหาก DIET พบโปรแกรมประเภทนี้แล้วก็จะข้ามไฟล์พวกนี้ไปโดยจะแจ้งข้อความให้เราทราบ เราสามารถที่จะบังคับให้ DIET ลองบีบขนาดโปรแกรมเหล่านี้ได้โดยใช้พารามิเตอร์ –i เช่น

```
C : > DIET -i Temp.EXE
```

ภาษาญี่ปุ่น

เนื่องจากคนเขียนโปรแกรม DIET เป็นคนญี่ปุ่น ดังนั้นหากเรามี JDOS ซึ่งเป็น PC-DOS ที่ทำงานเป็นภาษาญี่ปุ่น เราสามารถใช้พารามิเตอร์ –j เพื่อให้ DIET โต้ตอบกับเราเป็นภาษาญี่ปุ่น สำหรับท่านที่ใช้ JDOS 4.0 จากบริษัท IBM Japan อาจจะได้คำแนะนำเพิ่มเติมจาก Mr.Matsumoto บ้างแล้ว

สั่งให้ DIET ให้เงียบ

ในบางครั้งหากเราทำงานเป็น MultiTask หรือรันโปรแกรมบางตัวซึ่ง Message ของ DIET จะไปทำการรบกวนทำให้ข้อความบนจอดูไม่ค่อยรู้เรื่อง เราสามารถสั่งให้ DIET ทำตัวเงียบ ๆ ไว้โดยไม่ส่งแมจเสจโต้ตอบออกมาด้วยพารามิเตอร์ –m เช่น

```
C : >DIET - m TEST.COM
```

และด้วยคำสั่งนี้ DIET จะทำการบีบขนาดข้อมูลของไฟล์ TEST.COM แต่จะไม่มีข้อความต่าง ๆ ปรากฏขึ้นมาบนจอ

ให้ DIET เปลี่ยนแอตทริบิวต์

ดังที่ได้กล่าวไปข้างต้นว่าข้อมูลที่ถูกบีบให้เล็กลงไม่สามารถที่จะทำแก้ไขได้ หรือหากข้อมูลมีการแก้ไขและเก็บลงไปในชื่อไฟล์เดียวกัน ไฟล์ที่ถูกเก็บลงไปจะเป็นไฟล์ธรรมดาที่ไม่มีการบีบขนาดข้อมูลทันที เพื่อป้องกันความผิดพลาดที่อาจจะเกิดขึ้น ให้ใช้พารามิเตอร์ –a เพื่อสั่งให้ตัวมันเองเปลี่ยนแอตทริบิวต์ของไฟล์ที่บีบขนาดข้อมูลเสร็จแล้วเป็น Read-Only การใช้งานเหมือนกับพารามิเตอร์ –m

ทั้งหมดนี้เป็นพารามิเตอร์ที่ DIET มีให้ใช้ นอกจากพารามิเตอร์ดังกล่าวแล้ว DIET ยังสามารถที่จะกำหนดการทำงานได้จาก DOS ENVIRONMENT อีกด้วย, การ SET DOS ENVIRONMENT นั้นสามารถทำได้ด้วยคำสั่ง SET ซึ่งเป็น Internal Command ของดอสอยู่แล้ว มีคำสั่งใน ENVIRONMENT อยู่สองตัว ซึ่งทั้งสองตัวนี้มีไว้ใช้งานกับ DIET ที่ทำงานในแบบเรสซิเดนต์เท่านั้น คำสั่งแรกคือ ENVIRONMENT แรกคือ DIETOFF ซึ่งมีรูปแบบการใช้งานคือ

```
C:>SET DIETOFF=[filename.ext];
[filename.ext];...
```

Environment ตัวนี้จะสั่งให้ DIET ไม่มีการขยายข้อมูลให้กับโปรแกรมที่ถูกระบุ เช่น หากเราต้องการขยายขนาดของข้อมูลให้กลับสู่ปกติ นอกจากใช้คำสั่ง –r, –ra ของ DIET แล้ว เรายังสามารถที่จะคืนขนาดของข้อมูลที่ถูกบีบขนาดไว้ได้โดยใช้คำสั่ง COPY ธรรมดาของดอส หากเราให้ DIET ทำงานแบบเรสซิเดนต์แล้วใช้คำสั่ง เช่น

```
C:>COPY Compress.Dat Normal.Dat
```

ไฟล์ Normal.Dat ที่ได้จะเป็นข้อมูลธรรมดา (สามารถจะตรวจสอบได้โดยการ Dir ขึ้นมาดูขนาดของไฟล์) ดังนั้นหากมีการใช้คำสั่ง

```
C:>SET DIETOFF=Command.
com;xcopy.exe
```

โปรแกรม DIET จะไม่มีการคืนสภาพข้อมูล หากพบว่าโปรแกรมที่กำลังเรียกใช้ข้อมูลถูกบีบขนาดได้ระบุอยู่ใน Environment DIETOFF ด้วยเหตุนี้หากเราใช้คำสั่ง Copy ดังบรรทัดข้างบนอีกครั้งจะพบว่าไฟล์ Normal.Dat ที่ได้จะมีขนาดและข้อมูลเหมือนกับไฟล์ Compress.Dat ทุกประการ สำหรับคำสั่ง Copy นั้นเนื่องจากเป็น Dos Internal Command ซึ่งบรรจุอยู่ใน Command.com จึงจำเป็นต้อง DIETOFF Command.com ซึ่งมีผลข้างเคียงไปสู่คำสั่ง TYPE ของดอสเช่นกัน แต่สำหรับ Utility ใกล้เคียงคำสั่ง TYPE เช่น Browse หรือ List ซึ่งทำงานได้ดีกว่า คำสั่ง Type ยังคงทำงานได้ตามปกติ (หากเราไม่ไป DIETOFF มันเอาไว้)

คราวนี้คำสั่งหรือ Environment ตัวที่สองก็คือ DIETSIZE ซึ่งมีรูปแบบการใช้งานเหมือน DIETOFF คือ

```
C:>SET DIETSIZE=[filename.ext];
[filename.ext];...
```

Environment นี้จะทำให้ DIET ส่งขนาดไฟล์ที่แท้จริง (ขนาดของไฟล์ก่อนที่จะถูกบีบขนาด) ออกมาให้โปรแกรมต่าง ๆ หากโปรแกรมนั้นมีการเรียกใช้ Function Call หมายเลข 4Eh ของดอสที่ต้องมี Environment นี้เพราะว่ามีไวรัสระบาดอยู่ทั่วเมือง โปรแกรมต่าง ๆ สมัยนี้มักจะให้ตรวจสอบขนาด

ของตัวเองว่ายังปกติดีอยู่หรือเปล่า หากไม่ตรงกับที่เคยเป็นแล้ว อาจจะไม่ให้โปรแกรมทำงาน ดังนั้นหากเรากำหนด DIETSIZE เอาไว้ DIET ก็จะจัดการแจ้งขนาดไฟล์ดั้งเดิมของไฟล์นั้น ๆ กลับไปให้โปรแกรมที่ถามมา ตัวอย่างเช่น

```
C:>SET DIETSIZE=*.COM;*.EXE
```

แล้วมีการสอบถามขนาดของไฟล์ซึ่งมีนามสกุลตรงกับ .COM หรือ .EXE DIET จะจัดการส่งขนาดของไฟล์ที่แท้จริงกลับไป เมื่อ SET DIETSIZE แล้วไม่ต้องห่วงว่าจะไปรบกวนการทำงานของคำสั่ง DIR เลย เพราะว่าคำสั่ง DIR นั้นแสดงขนาดไฟล์เองโดยตรงไม่ได้เรียกผ่านฟังก์ชัน Call หมายเลข 4Eh ของดอสเลย

ข้อแนะนำบางประการ

ในบางครั้งหากเราพบ Error ว่า File not found หรือ File open Error เช่น "Can't find xxxx.ext" ทั้ง ๆ ที่มีไฟล์นั้นอยู่ในไดเรคทอรี เหตุการณ์นี้เกิดขึ้นได้นั้นเป็นเพราะว่า DIET ไม่สามารถสร้างไฟล์ชั่วคราวขึ้นมาได้ โดยปกติแล้ว DIET จะสร้างไฟล์ชั่วคราวที่ไดรฟ์และไดเรคทอรีปัจจุบัน แต่หากว่าเนื้อที่ในไดรฟ์นั้นหรือจำนวนไฟล์ในไดเรคทอรีนั้นเต็มจนไม่สามารถสร้างไฟล์ชั่วคราวดังกล่าวได้แล้วก็จะเกิด Error ดังที่เห็นการแก้ไขปัญหานี้ทำได้ง่ายมาก เพียงแต่ใช้พารามิเตอร์ -t เพื่อเปลี่ยนไดรฟ์หรือไดเรคทอรีที่ DIET จะสร้างไฟล์ชั่วคราวไปอยู่ที่อื่นที่ DIET สามารถสร้างไฟล์ชั่วคราวของมันได้ ปัญหานี้ก็จะหายไป

และในบางครั้งหากพบไฟล์ "-DIET__.***" ไม่ต้องแปลกใจ เพราะ

ไฟล์นี้เป็นไฟล์ชั่วคราวที่ถูกสร้างขึ้นโดย DIET แต่ก่อนที่จะถูกลบออกไป เครื่องอาจจะถูก Reset หรือไฟฟ้าดับ ทำให้มันยังคงค้างอยู่ในไดเรคทอรี

FreeWare

คราวที่แล้วถ้าจำได้ผมเขียนบทความ 800K ได้พูดถึง ShareWare ไปแล้ว มาคราวนี้ผมมาพูดถึง FreeWare กันบ้าง ตามที่ดูชื่อแล้วก็คงจะเดาออกว่า FreeWare คือโปรแกรมที่คนเขียนสร้างขึ้น เพื่อแจกจ่ายให้ผู้อื่นใช้กันฟรี ๆ และสามารถก๊อบปี้แจกกันได้อย่างไม่ผิดกฎหมายลิขสิทธิ์แต่อย่างใด สำหรับ DIET ก็เป็น FreeWare ตัวหนึ่งเช่นกัน ครั้งแรกที่ทราบว่า DIET เป็นโปรแกรมประเภท FreeWare ทำเอาช็อคอยู่พอสมควร เนื่องจากโปรแกรมนี้มีประสิทธิภาพในการทำงานสูงและตามความรู้สึกของผม ความสามารถสูงกว่าโปรแกรมที่ทำขายกันซะด้วยซ้ำไป ต้องขอขอบคุณ Mr.Teddy Matsumoto จริง ๆ ที่สร้างโปรแกรมนี้ให้เราทุกคนได้ใช้กัน อย่างไรก็ตาม ถึงแม้ว่า FreeWare ถูกสร้างขึ้นมาเพื่อแจกกันฟรีก็ตาม แต่ก็มีวัฒนธรรมบางอย่างที่เราควรจะปฏิบัติต่อ FreeWare เพื่อเป็นการให้เกียรติแก่ผู้เขียนโปรแกรมเหล่านี้โดยการ

1. ไม่นำมาก๊อบปี้เพื่อจำหน่าย
2. ไม่นำมาใช้ในงานธุรกิจ เพื่อหารายได้เข้าสู่ตนเอง
3. เมื่อก๊อบปี้แจกจ่าย ควรจะกระทำให้ครบทุกไฟล์ตามต้นฉบับที่ ผู้เขียนโปรแกรมได้สร้างเอาไว้

สำหรับผู้ที่ต้องการรายละเอียดต่าง ๆ จากคุณ Teddy Matsumoto โดยตรงก็สามารถส่ง Electronic Mail ไปที่ GBG00230 Nifty Serv แต่สำหรับผู้ที่อาศัยอยู่ในประเทศไทย สามารถติดต่อได้โดยผ่านคุณ K.Okubo ที่ 74100, 2565 Compuserve, K.Okubo Genie, หรือ C00236@ sinet.ad.jp โดยผ่านเครือข่าย FidoNET จากเมืองไทย

รายละเอียดการใช้โปรแกรม DIET สามารถที่จะอ่านเพิ่มเติมได้จากไฟล์ DIET.DOC และสำหรับผู้อ่านที่มีข้อสงสัยหรือต้องการติชม สามารถติดต่อผ่านทางวารสารไมโครฯ หรือโทรมาหาผมโดยตรงก็ได้ 375-4588 สำหรับท่านที่มีโมเด็มสามารถติดต่อโดยตรงที่ Pjk BBS หรือจะส่งเป็น Net Mail ในระบบ Fido NET ถึง Prajak Sk. แล้วส่งมาที่ Address 6:608/19 ก็ได้เช่นกันครับ สวัสดีครับ!